

Introduction to Semantic MediaWiki

Yaron Koren
SMWCon Spring 2015
St. Louis, Missouri
May 6, 2015



About me

- MediaWiki extension developer / entrepreneur / author
- Based in New York City
- Involved with MediaWiki/SMW since 2006



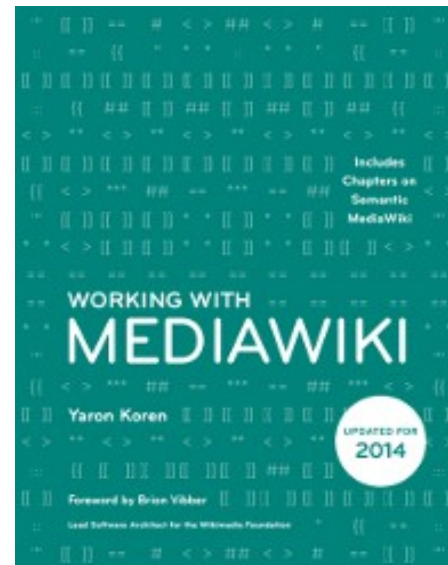
My company: WikiWorks

- MediaWiki development and consulting
- <http://wikiworks.com>



My book: Working with MediaWiki

- The only MediaWiki reference guide since 2007
- Just updated last year!
- <http://workingwithmediawiki.com>



Our wiki farm: Referata

- Free and paid MediaWiki/SMW hosting
- <http://referata.com>



My development work

- I have written 16+ MediaWiki extensions, most notably Semantic Forms
- My most recent extension is Cargo, which is... er... um... er.... an alternative to Semantic MediaWiki.
- But we won't talk about that today!



Semantic MediaWiki

An open-source extension to MediaWiki that lets you store and query data within the wiki's pages.

Not the most popular extension, but the one that's the most a “brand”.



SMW is unique

One of few MW extensions with its own:

- Mailing list
- Spinoff extensions (~100 total, ~50 active)

The only MW extension with its own:

- Website (semantic-mediawiki.org)
- Social media presence
- Conference (SMWCon!)



History of SMW

- Conceived of in 2005 by Denny Vrandečić and Markus Krötzsch – two German grad students with a Semantic Web focus



SMW: Conceived as a Wikimania talk

- The 1st Wikimania was held in Frankfurt, Germany in August 2005
- Markus and Denny's talk: “Wikipedia and the Semantic Web”
- Their proposal: allow for tags in Wikipedia, of the form “[[property::value]]”, that store semantic triples for (a) RDF export and (b) internal use.

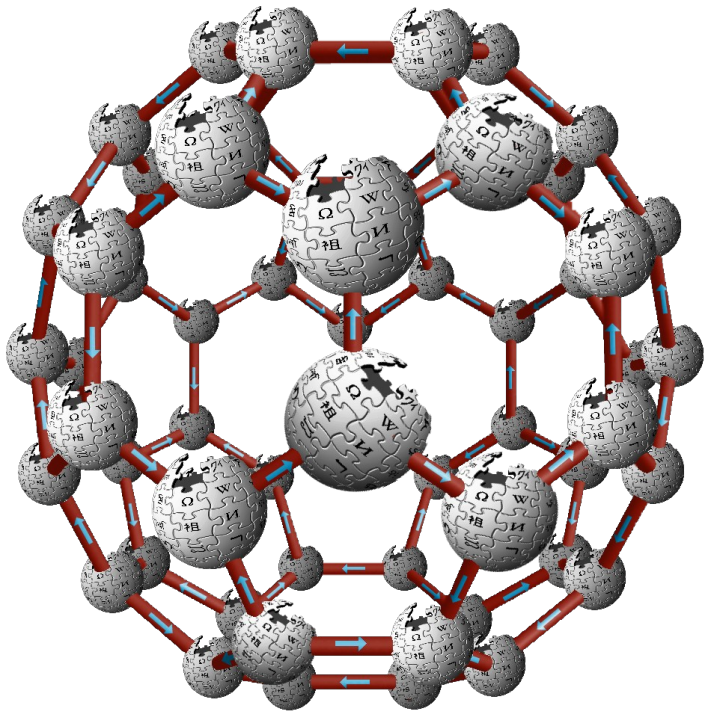


Response was immediate!

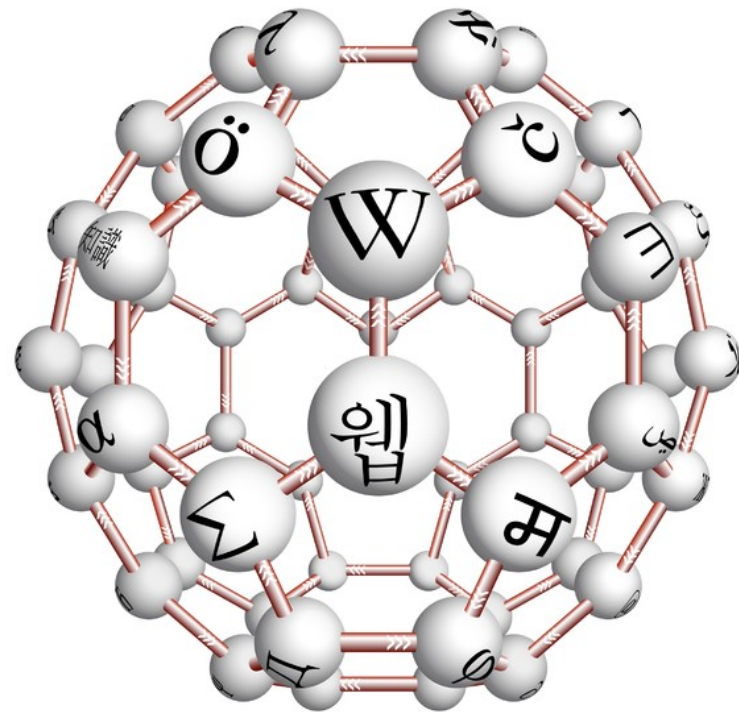
- Markus and Denny got funding from various sources to develop the concept into a MediaWiki extension
- “Semantic MediaWiki” was first released on September 30, 2005



First SMW logos



2006



2007



Then a funny thing happened...

Wikipedia/Wikimedia ignored SMW, but *organizations* that used MediaWiki embraced it.



Where do I come in?

- In 2006, I and a friend decided to develop a site like Rotten Tomatoes, but for political opinions (not movie reviews)
- We quickly decided on MediaWiki/SMW as the tool
- Discourse DB (discoursedb.org) was released in September 2006



Our 1st innovation

- All semantic data should be stored in templates – and most templates should hold semantic data
- No free-form tagging like “He lives in [[Lives in::Europe]]”.



My 2nd innovation

- A form! I created a custom form that generated pages that called the site's main template.
- (My friend left the project by then)



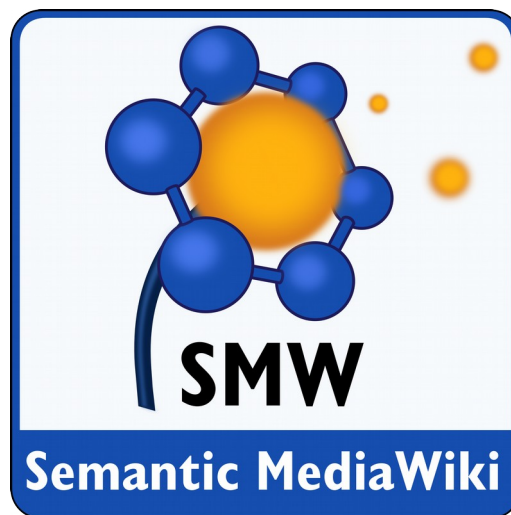
My 3rd innovation

- “Semantic Forms”, an extension that lets people create their own forms, to create and edit pages holding structured data
- First released on May 2, 2007



SMW's current status

- In use on hundreds (thousands?) of wikis
- At version 2.1.3
- Current main developers are Jeroen De Dauw and “MWJames”; Markus contributes a little



What about “Semantic Wikipedia”?

- Denny started Wikidata, which launched in October 2012
- Wikidata: inspired by SMW, but (a) supports multiple languages, and (b) requires a reference for every fact
- A massive project: currently holds millions of facts
- Currently only used on Wikipedia for interlanguage links.



Storing data in SMW

Three ways:

- `[[Lives in::United States]]`
- `{{#set:Lives in=United States}}`
- `{{#subobject:-|Lived in=United States|Start year=2000|End year=}}`



#subobject

- The subject of the triple (not the object – yes, confusing) is an abstract entity within the page, not the page itself
- Allows for defining a table of data within a page – each #subobject call is one row
- The Semantic Internal Objects extension does the same thing (SIO was first)



Wait, aren't we storing all data in templates?

Yes, okay:

- `[[Lives in::{{{Country|}}}]`
- `{{#set:Lives in={{{Country|}}} }}`
- `{{#subobject:-|Lived in={{{Country|}}} |Start year={{{Start year|}}} |End year={{{End year|}}} }}`

(#subobject calls should be made from multiple-instance templates)



Defining a property

- Each property gets its own page, at `Property:property-name`
- Properties are defined with a type, using “`[[Has type::type-name]]`”
- Types: Text, Number, Date, Boolean, Quantity, Geographic coordinate, URL, Email, Code, Record...



“Enumerated” properties

- Use “Allows value” to set a pre-defined group of allowed values

This is a property of type `[[Has type::Text]]`.

The allowed values are:

- * `[[Allows value::Low]]`
- * `[[Allows value::Medium]]`
- * `[[Allows value::High]]`



Let's create some data!

- Good starting page to know about:
Special:CreateClass (actually defined by Semantic Forms)
- <http://smwconstlouis.referata.com/>
- Please go there, create an account, then use the links in the sidebar.



Querying data

Two ways: #ask and #show

- `{{#ask:[[Category:People]]|?Lives in}}`
- `{{#show:John Doe|?Lives in}}`

The first displays a table, the second displays a single value



#ask queries can be formatted

- Using “format=” parameter
- Defined formats: list, ol, ul, table, broadtable, category, template, embedded, count
- Also, export formats: csv, dsv, json, rss, rdf
- Each format can define its own additional parameters (like “width=”)



Many more formats

- Semantic Result Formats extension defines formats for timelines, calendars, charts, graphs, math functions, outline display, etc.
- Semantic Maps extension defines formats for map display
- Other extensions define even more formats



Special:Ask

- Provides a UI to create an #ask query, and view its results
- The easiest way to see the set of available options on a particular wiki



Parts of a query

```
{ {#ask:  
  [[Category:People]] ← “filter”  
  |?Lives in           ← printouts  
  |?Occupation  
  |mainlabel=Name      ← standard  
  |limit=20            params  
  |format=template  
  |template=PersonFormatter} } ← format param
```



Other SMW special pages

- `Special:Properties`, `Special:UnusedProperties`, `Special:WantedProperties` – lists of properties
- `Special:Browse` – shows the set of property/value pairs for any page
- `Special:SearchByProperty` – search for all pages with a specific property/value combination (rarely used?)



Page metadata

- Various special properties to encode “metadata”:
 - Creation date
 - Modification date
 - Last editor is
 - etc.

